|  |  |  |
| --- | --- | --- |
| **Tech Saksham**  Final Project Report  **Track Name** |  |  |

**“Automatic question paper generation.”**

**“** **Avanthi Institute of Engineering and Technology”**

|  |  |
| --- | --- |
| **ROLL NO** | **NAME** |
| 19Q61A0564 | KANDHI VIDHYA REDDY |
| 19Q61A0530 | JUVVANTHULA HARSHA |
| 19Q61A0598 | THATI DEEPIKA |
| 19Q61A0571 | KOLI HARINA |
| 19Q61A05D8 | THEDLA PRAVALIKA |

|  |  |
| --- | --- |
|  |  |
|  | Trainer Name: UMA MAHESHWARI |
|  | Master Trainer |

**ABSTRACT**

This is a challenging era due to the growth in the field of computer science and demand we are facing today. Hence examinations play a vital role in testing student’s performance. And that is why it is important to have a smart development question model for growth of students as well as to test their learning skills thereby keeping a check on student performance. Now the traditional method of generating question paper has been manual. In this method certain officials check out the question paper. But this method can be ineffective at times, repetition and security concerns. We have proposed an automated process of Question Paper Generation which is fast, streamlined, randomized and secure. Every task performed by this system is automated so that storage space, bias and security is not a concern anymore. Furthermore, we have proposed a new algorithm which ensures total randomization of questions and avoids repetitions. The proposed system can be helpful to many educational institutes and NGO based institutes.

**SYSTEM ANALYSIS**

# EXISTING SYSTEM

**2.1 EXISTING SYSTEM**

The existing system for Question Paper Generation requires human staff to chalk out questions that appear in the question paper. These teachers or professors select the questions according to the syllabus and pattern as prescribed by the curriculum. The question paper then may be referred to a higher authority who has the final say in these matters.

**DISADVANTAGES**

1. Prone to damage
2. Inefficient document transportation
3. Supply costs
4. Poor environmental credentials
5. Limited collaboration
6. Editing problems

# PROPOSED SYSTEM

The paper on „Automatic Question Paper Generation System using Randomization Algorithm‟ describes a system which uses a shuffling algorithm (existing algorithm) as a randomization technique. The system defines several modules such as user administration, subject selection, difficulty level specification, question entry, question management, paper generation, and paper management

**ADVANTAGES**

1. User can generate test papers randomly and instantly, thus saving a lot of time.
2. The algorithm enables randomization of questions.
3. A new question can be added to the database at any instance and different sets of test papers could be generated without any limitation

**CHAPTER 1**

**INTRODUCTION**

In today’s world, time is a major concern. Any product that can effectively reduce time and power consumption is accepted and appreciated. Thus we are presenting an Automated Question Paper Generator System that can reduce time consumption by replacing the conventional method of question paper generation system. It also needs lesser man power. We have implemented a role-based hierarchy which restricts access to the users. The system also deploys security mechanisms that prohibit duplication of question papers. There are provisions to enter and edit data suitable to any educational organization with complete freedom for specifying courses, semesters, syllabus and pattern. This enables an educational institute to generate question ensuring security and non-repetitiveness of question papers and is a boon for organizations with limited staff and resources. Our system aims to provide fast operations, data storage and high security for all its tasks. The Question Paper Generator system automatically generates paper, prepares doc file as per selected paper format. Also emails it to other colleges. After this question paper is converted to pdf file and emailed to colleges on button click. A Automation means to replace the manual operations with computer procedures and other machines. Automation is aimed at increasing productivity, manufacturing prowess. It also reduces costs, labor and eliminates human error. B. Types of Automation helps to increase productivity and reduces costs in industries. Automation plays a crucial role in manufacturing industries. Automation can be of different types: - a. Information technology (IT) b. Computer-aided manufacturing (CAM) c. Numerically controlled (NC) equipment d. Robots e. Flexible manufacturing systems (FMS) f. Computer integrated manufacturing (CIM)

**CHAPTER 2**

**SERVICES AND TOOLS REQUIRED**

**HARDWARE REQUIREMENTS**

# Processor - Intel

* Speed - 1.1 Ghz
* RAM - 2 GB,4 GB
* Hard Disk - 500 GB
* Key Board - Standard Windows Keyboard
* Mouse - Two or Three Button Mouse
* Monitor - LCD

**SOFTWARE REQUIREMENTS**

Operating System : Windows Family

* Programming Language : PHP, Java Script
* Database : My Sql
* Web Server : XAMPP Server
* Tools : Sublime Text, Navicat Lite 8.2

**CHAPTER 3**

**SYSTEM DESIGN**

**UML DIAGRAMS**

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

**GOALS:**

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

**USE CASE DIAGRAM:**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

****

****

**CLASS DIAGRAM:**

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

****

**SEQUENCE DIAGRAM:**

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



**Collaboration:**



**CHAPTER 4**

**MODULES**

* User
* Admin

**DESCRIPTION**

1. **Admin**

here admin is a module, admin can directly login with the application, after successful login he can perform some operations such as add questions and view questions and logout.

1. **User**

In this application user is also a module, if user want to access our service, he/her must register with the application and after success login only, use can access perform some operations such

Select subject and enter how many questions you want to generate and click on generate button to generate paper and logout.

**SYSTEM TESTING**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

**TYPES OF TESTS**

**Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**Functional test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**Unit Testing:**

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

**Test strategy and approach**

Field testing will be performed manually and functional tests will be written in detail.

**Test objectives**

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

**Features to be tested**

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page.

**Integration Testing**

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**Acceptance Testing**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**CONCLUSION**

In this paper, an automated Question Paper Generator is proposed which is implemented as a real-time application in Barshi, Solapur. The proposed work describes an automated system that progresses from the traditional method of paper generation to an automated process, by providing controlled access to the resources. We have also considered the importance of randomization in the task of paper generation. Our system has deployed an efficient algorithm which is totally randomized and avoids repetition of questions is consequent question papers, making it impossible to derive any pattern in the papers. We distinguish between administrators and subordinates by their tasks. Therefore, the resultant automated system for Question Paper Generation provides improvement in terms of controlled access to the resources, random generation of question papers and a secure platform.

**REFERENCES**

[1] Automated Question Paper Generation System Rohan Bhirangi, Smita Bhoir Computer Engineering Department, Ramrao Adik Institute of Technology Navi Mumbai, Maharashtra, India April 2016 ISSN: 2278-9359 (Volume-5, Issue-4)

[2] Surbhi Choudhary, Abdul Rais Abdul Waheed, Shrutika Gawandi and Kavita Joshi, “Question Paper Generator System,” International Journal of Computer Science Trends and Technology, vol. 3, issue 5, Sept – Oct 2015.

[3] Dan Liu, Jianmin Wang and Lijuan Zheng, “Automatic Test Paper Generation Based on Ant Colony Algorithm,” Journal of Software, vol. 8, no. 10, October 2013.

### 